Performance Comparison of Three Routing Protocols for Ad Hoc Networks

Hong Jiang
J. J. Garcia-Luna-Aceves
Computer Engineering Department
Baskin School of Engineering
University of California
Santa Cruz, CA95064, USA
hjiang,jj@cse.ucsc.edu

***Abstract*— Many routing protocols for ad hoc networks have been proposed to date. Among them, STAR is a representative table-driven protocol, while AODV and DSR are two representative on-demand protocols. This paper analyzes these three protocols using the GloMoSim simulation environment. The scenarios used in the simulation experiments take into account a variety of environmental factors that influence protocol performance. The performance of the protocols is compared in terms of their control overhead, amount of data delivered, and average latency in packet delivery. The simulation results show that STAR achieves better overall performance than AODV and DSR in sparsely connected networks. For the case of densely connected networks, AODV performs better in terms of data delivery, while STAR performs much better in terms of control overhead. The study also addresses the question of how accurate a simulator could be regarded for presenting the characteristics of the routing protocols and for comparison purposes.**

I. INTRODUCTION

An ad hoc network is a group of wireless mobile devices (nodes) that communicate with each other in a collaborative way, over multi-hop wireless links, without any stationary infrastructure or centralized management. Examples of ad hoc networks include: disaster situations such as earthquake and flooding, where the rescue teams need to coordinate themselves without the availability of fixed networks; soldiers in a battlefield exchanging tactical information; entrepreneurs in a meeting sharing business information [1].

The high mobility and low bandwidth features of ad hoc networks make it necessary for a routing protocol to be dynamic and bandwidth efficient to enable the delivery of data packets while producing low control overhead. Such traditional routing protocols as OSPF [2] designed for wired networks do not met such requirements.

Many routing protocols have been proposed for ad hoc networks [3], [4], [5], [6], [7], [8], [9]. The mechanisms they adopt were traditionally categorized as table-driven and on-demand. On-demand routing protocols query a route when there is a real need (demand) for it. In contrast, table-driven routing protocols maintain routing information for all network destinations independently of the traffic to such destinations.

Several performance comparisons have been reported for ad hoc routing protocols in the recent past [10], [11], [12], [13]. This paper compares the Ad Hoc On-Demand Distance Vector protocol (AODV [5], [14]) and the Dynamic Source Routing protocol (DSR [6]), with the Source Tree Adaptive Routing protocol (STAR[4]). AODV and DSR are the two most popular on-demand protocols to date, while STAR is a representative table-driven protocol for ad hoc networks environment. The comparison is made in terms of data delivery, control overhead, and average latency, using the GloMoSim simulation environment [15].

This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under grant F30602-97-2-0338.

Section II reviews the key features of the three routing protocols under study. Section III describes the common simulation environment and implementation parameters for the protocols. Section IV presents the factors considered in designing the simulation scenarios. Section V presents the results of the simulation study. The results from five different scenarios show that STAR provides the best performance among the three protocols analyzed for the case of sparsely connected topologies, while AODV provides the best data delivery and STAR incurs the least amount of overhead with slightly worse data delivery than AODV in densely connected topologies. Interestingly, the simulation results for AODV and DSR in GloMoSim differ in some respects from published results using ns-2. The results show that STAR faces a scaling problem as the number of network nodes grows larger, while on-demand routing protocols face scaling problems as the number of flows per node grows, specially if the number becomes proportional to the number of nodes in the network, which suggests the need for a hybrid approach to routing in ad hoc networks.

II. ROUTING PROTOCOLS

A. STAR

STAR [4] is a table-driven routing protocol. Each node discovers and maintains topology information of the network, and builds a shortest path tree (source tree) to store preferred paths to destinations. The basic mechanisms in STAR include the detection of neighbors and exchange of topology information (update message) among nodes.

For STAR, there are mainly two alternative mechanisms to discover neighbors:

1. *Hello Messages:* Hello messages are sent by each node periodically to inform neighbors of its existence. Such messages can be small packets, not needing to contain any routing information. When a node receives a hello message from another

Report Documentation Page					*Form Approved* *OMB No. 0704-0188*
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE **2001**		2. REPORT TYPE			3. DATES COVERED **00-00-2001 to 00-00-2001**
4. TITLE AND SUBTITLE **Performance Comparison of Three Routing Protocols for Ad Hoc Networks**					5a. CONTRACT NUMBER
					5b. GRANT NUMBER
					5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)					5d. PROJECT NUMBER
					5e. TASK NUMBER
					5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California at Santa Cruz,Department of Computer Engineering,Santa Cruz,CA,95064**					8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)					10. SPONSOR/MONITOR'S ACRONYM(S)
					11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**					
13. SUPPLEMENTARY NOTES **The original document contains color images.**					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES **8**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

node that it does not know previously, it discovers a new neighbor. If a node does not receive any message (update or hello) from a neighbor for a certain period of time, it determines that this neighbor is broken or out of its range.
2. *Neighbor Protocol:* A neighbor protocol can be implemented at the link layer. It notifies STAR of the existence of new neighbors or the loss of connectivity to an existing neighbor. With the support of a neighbor protocol, no hello messages are needed.

By adopting the Least-Overhead Routing Approach (LORA), STAR greatly reduces control overhead in ad hoc network environment. Under LORA, a source node does not need to maintain shortest paths to destinations. A node running STAR does not send update messages after every change of topology. It only sends updates in the event of unreachable destinations, new destinations, the possibility of permanent routing loops, or cost of paths exceeding a given threshold. These situations are defined by the three basic LORA rules in STAR.

Four LORA rules are further defined for the case when the underlying MAC protocol does not support reliable transmission. These rules introduce periodic update messages, repair messages and query messages. Query messages give some on-demand characteristic to STAR, but they are used much less aggressively than in such on-demand protocols as AODV and DSR.

The basic information unit in STAR is the representation of a link, which indicates the two adjacent neighbors, the cost of the link, and the time stamp reflecting the freshness of the link. Accordingly, for communicating topology information, the basic information unit transmitted is a LSU (Link State Update). The set of links used by a node in its preferred paths to destinations form the source tree of the node. The set of LSUs form the topology information being exchanged.

B. AODV

A node running AODV [14] initiates a route discovery process only when it has data packets to send and it does not know any route to the destination node, that is, route discovery in AODV is “on-demand”.

During a route discovery process, the source node broadcasts a route query packet to its neighbors. If any of the neighbors has a route to the destination, it replies to the query with a route reply packet; otherwise, the neighbors rebroadcast the route query packet. Finally, some query packets reach the destination, or nodes that know a route to the destination. At that time, a reply packet is produced and transmitted tracing back the route traversed by the query packet. To handle the case in which a route does not exist, or the query or reply packets are lost, the source node rebroadcasts the query packet if no reply is received by the source after a time-out.

A path maintenance process is used by AODV to monitor the operation of a route being used. If a source node receives the notification of a broken link, it can re-initiate the route discovery processes to find a new route to the destination. If a destination or an intermediate node detects a broken link, it sends special messages to the affected source nodes.

AODV uses a routing table to specify distances to destinations. It uses sequence numbers maintained at each destination to determine the freshness of routing information and to prevent routing loops. It uses timers to monitor the utilization of routing information. A routing table entry is “expired” if not used for a period of time.

The recent specification of AODV [14] suggests an optimization to AODV: it uses an expanding ring search to discover routes to an unknown destination. In the expanding ring search, increasingly larger neighborhoods are searched to find the destination. The search radius is controlled by the TTL field in the IP header of the request packets. If the route to a previously known destination is needed, the prior hop-wise distance is used for the radius.

C. DSR

DSR [6], [16] adopts a similar on-demand approach as AODV regarding the route discovery and maintenance processes. A key difference of DSR from AODV and other on-demand protocols is the use of source routing, where the source node specifies the complete sequence of intermediate nodes for each data packet to reach its destination. The source-route information is carried by the header of the data packet. The advantage of source routing is that no additional mechanism is needed to detect routing loops. The obvious disadvantage is that data packets must carry source routes.

The data structure DSR uses to store routing information is route cache, with each cache entry storing one specific route from the source to a destination.

DSR makes very aggressive use of the source routing information. Every intermediate node caches the source route carried in a data packet it forwards, and the following optimization rules to DSR have also been proposed:
1. *Salvaging:* If an intermediate node discovers that the next hop in the source route is unreachable, it can replace the source route in the data packets with a route from its own cache.
2. *Gratuitous Route Repair:* A source node notified error of the packets it originates propagates the error notification to its neighbors by piggy-backing it on its next route request. This helps clean up the caches of other nodes in the network that may have the failed link in one of the cached source routes.
3. *Promiscuous Listening:* When a node overhears a packet that is addressed to another node, it adds the source route information into its own route caches. The node also checks if the packet could be routed via itself to gain a shorter route.

III. Implementing Protocols in GloMoSim

GloMoSim [15] is a library for simulating wireless networks. It was developed using PARSEC [17], a C-based parallel simulation language. In GloMoSim, the library structure is decomposed into different network layers. A number of protocols have been developed at each layer. New protocols and

modules can be programmed and added to the library using PARSEC. A strong point of GloMoSim, compared with several other wireless network simulators, is the capability to simulate very large networks with thousands of nodes.

To conduct this performance study, we developed an implementation of STAR in GloMoSim and used the implementations of AODV and DSR already available in the GloMoSim library.

All three routing protocols use the network layer service to communicate control messages with neighbors. On the other hand, when delivering data packets, the network layer calls the routing protocol to determine the paths to the destinations. If no path is available for a destination, the source node queues the data and send a query message to its neighbors. If no reply to this query is received after a time-out period, another query will be sent, with a longer time-out period.

Practically, a timer should be set to determine whether a data packet has been queued too long, and in this case to drop the data packet. In our study, for the purpose of comparison, none of the simulated protocols set such a timer.

The routing layer is notified by the MAC layer if a control or data packet can not be delivered to the next hop by the MAC layer after a given numbers of retries. The routing protocols would regard this as an indication of the loss of connection with the neighbor.

A routing protocol can choose to use the promiscuous mode supported by MAC layer. Promiscuous mode means that a node in a network accepts all packets, regardless of their destination addresses.

The following implementation parameters were adopted for each protocol:

1. *STAR:* Our implementation of STAR does not use hello messages. Instead, we utilize periodic routing update messages to discover neighbors and to refresh the topology graph at each node in the network. The update broadcast event is triggered about every 6 seconds. The jitter for the update timer is 1 second. The periodic updates also provide STAR an additional way to detect broken links, besides using the notification from the MAC layer. Because every node sends messages at least every 6 seconds, if a neighbor is not heard from after 36 seconds, the connection to this neighbor is regarded as broken. The time-out value for the initial query in STAR is 600 milliseconds. It increases by 10 times after a time-out. The maximum time-out period is fixed at 600 seconds. To determine the appropriate values of the parameters such as time-out period, we tried several different alternatives, but still the chosen values are not guaranteed to be optimal. A parameter value suitable for one scenario may not be suitable for another scenario. This situation may also happen in AODV and DSR.

2. *AODV:* Promiscuous mode is set in AODV implementation. The timer to check whether a route is too old is set to 10 seconds. The timer to check whether a query is answered is set to the product of 80 milliseconds and TTL, where TTL starts at 1, or the previously known hop count, and increased by 2 after each time-out, until reaching the expected maximum radius, which is set as 35 in the simulation. The jitter for broadcast messages is 10 milliseconds.

3. *DSR:* The time-out period to check the reply for a query is initially set to be 500 milliseconds, it doubles after each time-out until it reaches 10 seconds.

Besides choosing the routing protocols, we also need to determine the protocol for each layer of the network stack. "Free space" mode is chosen for the propagation mode. For the radio layer, we choose "no capture". The MAC layer protocol is 802.11 [18]. The network protocol is IP. For the transport layer we choose UDP protocol. The application for generating data traffic is CBR.

The choices made above are mainly based on previous simulation work, so that the results of this study could be compared with the results from previous studies.

IV. Environmental Factors

A. Degree of Connectivity among Nodes

In many scenarios simulated in previous simulation studies of ad hoc networks, nodes were usually densely connected. In a highly dense network, almost every node has at least a path to any other node, usually just a few hops away. Meanwhile due to the high volume of routing control messages, congestion happens frequently in such networks. A sparsely connected ad hoc network bears different characteristics. In such a network, paths between two nodes do not always exist, and routing choices are more obviously affected by the mobility of the network.

In our simulation study, we ran simulations in both sparse and dense networks. Fixing the area to be 4km * 4km, and the number of nodes to be 40, the transmission range of each node in the sparse network is 250m, while in the dense network it is 400m.

B. Degree of Mobility

Varying the degree of mobility, or the moving speed of each node in the network, is a useful way to test how adjustable a routing protocol is to the dynamic environment. There have been several mobility models used in the past. We chose the "random waypoint" because this has been used more widely than other mobility models. In this model, each node begins the simulation by remaining stationary for "pause time" seconds. It then selects a random destination in the simulation space and moves to that destination at a speed distributed uniformly between a minimum and a maximum speed. Upon reaching the destination, the node pauses again for "pause time" seconds, selects another destination, and proceeds there as previously described, repeating this behavior for the duration of the simulation.

In our simulations, we fix the minimum moving speed to be 0, maximum speed to be 20m/sec. We varied the "pause time" between 0 and 900 seconds. A "pause time" of 0 second cor-

responds to continuous motion, and a pause time of 900 corresponds to no motion when the simulation time is 900 seconds.

C. Number and Duration of Data Flows

Because on-demand protocols query routes only when data flows exist for them, the number of data flows would influence the number of paths found and the control overhead for on-demand protocols, such as AODV and DSR. Because STAR also uses query messages in unreliable networks, we expect it to be also affected by this factor. How well a protocol adjusts to the change of data flows is another important criterion for evaluating a routing protocol. In our simulations, we varied the number of data flows to be 20 and 60.

In most previous simulation studies, each data flow started at an early time of the simulation period, and continued until almost the end of the period. In our simulations, besides this long lasting flow pattern, we also tested the protocols under data flows that last shorter time periods.

D. Shape of Space and Initial Node Placement

Different shapes of moving space would affect the mobility pattern of the nodes, and thus affect the simulation results. Square and rectangle with the similar area are expected to cause the routing protocols to work differently. Compared to a rectangle site with comparable area size, a square site models situations in which nodes can move more freely around each other. In our simulations, we chose both square and rectangle moving areas. But the rectangle area are not as narrow as what's chosen in [16]

The positions of the simulated nodes can be chosen uniformly or randomly, or be specifically defined for each. Since this factor might influence performance results, we chose both uniform and random node placement in the simulations.

E. Other Factors

There are also other factors for which we did not change the values and study the effects. We did not study the effect of having a static node or a few static nodes as points of attachments to the Internet, such that most of the traffic in the ad hoc network is to and from such point(s).

In our simulation and several previous simulations, traffic type was chosen to be constant bit rate source (CBR). In a real case, there are all kinds of popular applications with different traffic patterns from CBR. To observe the protocols more objectively, it is worth trying different applications in the future studies.

V. Results and Analysis

A. Metrics

The following metrics are used in the simulation study.

1. *Data Delivered:* The number of data packets delivered by all the nodes during the simulation period.
2. *Control Overhead:* The number of control packets sent by all the nodes to discover and maintain routes.

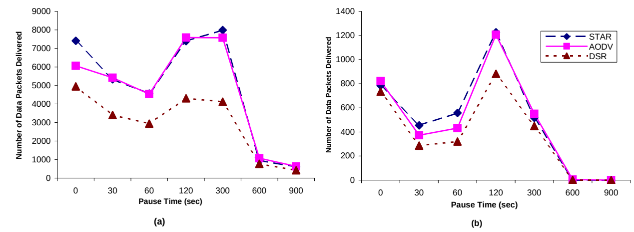

Fig. 1. Data Delivery in Scenario 1 for (a)60 (b)20 Data Flows

3. *Average Latency:* The average time delay between the time when a data packet is given to IP layer at the source node and the time when the packet arrives at the IP layer of the destination node.

These metrics are the most widely used for representing performance. When we design a routing protocol, higher data delivery, lower control overhead, and lower latency are always desired, but they can seldom come together. For example, when a protocol discovers more routes, it is also likely to produce more control packets.

The following sections present the results on the three metrics for 5 different scenarios. We also varied the number of data flows, and the value of the "pause time" (the time a node stays at a position before moving to the next random position) under each scenario.

B. Scenario 1

In this scenario, 40 nodes move in a 4km * 4km area. Nodes are initially placed uniformly within this area, and the power range for each node is 250m. As for the data flows, source and destination pairs are chosen randomly, with each flow lasts for most of the simulation time. The simulation time is 900 seconds. The network bandwidth is 2Mbit/sec.

Fig. 1, 2, and 3 show the results of the different metrics, under the combination of different degrees of mobility and numbers of flows.

Under the conditions of different mobility and different number of sources, STAR and AODV delivered comparable amounts of data packets. DSR had the lowest data delivery results.

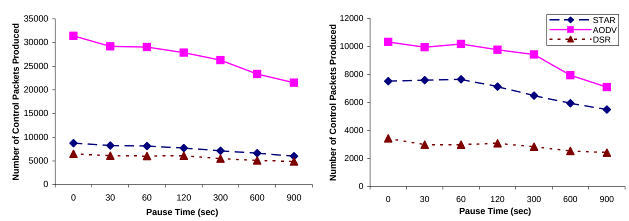

Fig. 2. Control Overhead in Scenario 1 for (a)60 (b)20 Data Flows

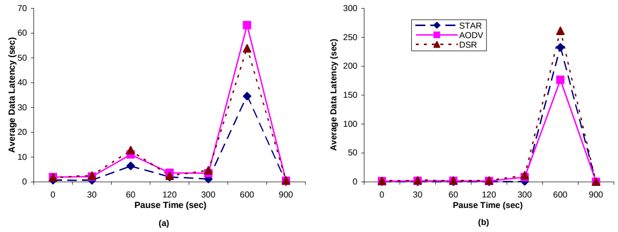

Fig. 3. Data Latency in Scenario 1 for (a)60 (b)20 Data Flows

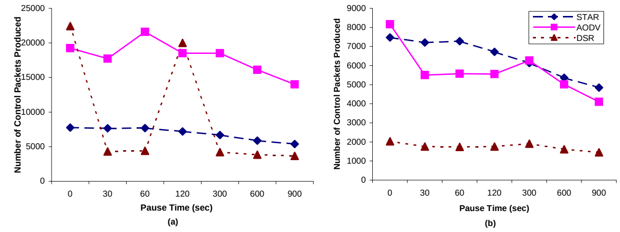

Fig. 5. Control Overhead in Scenario 2 for (a)60 (b)20 Data Flows

DSR produced the least amount of control overhead, while AODV produced the highest amount. The control overhead of STAR is comparable with DSR when the number of data flows is 60. AODV produced as much as five times control packets compared with DSR for the situation of 60 data flows. For STAR, the control overhead does not change much when the number of data flows changes, but for the other two protocols, especially for AODV, the overhead increases obviously when the number of data flows increases. This is also the typical difference between a table-driven protocol and an on-demand protocol.

STAR had the lowest average latency, while AODV and DSR had comparable latencies in overall. Overall, with a “pause time” of 600 seconds, the latency values for each protocol are much higher than in other situations, which shows that in this mobility mode the paths are not available at the early stage of the simulation.

C. Scenario 2

In this scenario, most environmental factors are the same as in Scenario 1, except that each data flow lasts for a shorter period of time.

Fig. 4, 5, and 6 show the results under this scenario.

In this scenario, STAR and AODV delivered comparable amounts of data packets, and DSR had the lowest data delivery results. At 20 data flows, the results for all the three protocols are comparable.

In the “data delivery” graph, when “pause time” decreases from 900 to 0 seconds, the trends the protocols follow are similar to the trends in Scenario 1, except for the case of a “pause time” equal to or less than 60 seconds. During this period, the three protocols deliver fewer data packets than in Scenario 1. This shows that when flows last shorter amounts of time, the delivery of data packets is negatively influenced, especially in cases with higher mobility.

DSR produced an abnormally high amount of control packets in several situations for 60 data flows.

Concerning average latency, the results are comparable for the three protocols. Again when the “pause time” is 600 seconds, the latency for each protocol is substantially higher than in other mobility situations.

D. Scenario 3

In this scenario, most environmental factors are the same as in Scenario 1. The only difference is that the area in which the nodes move is a rectangle with an of area 5km * 3km.

Fig. 7, 8, and 9 show the results for this scenario.

STAR is comparable with AODV in terms of data delivery. DSR had the lowest data delivery results. The behavior of the three protocols is interesting for 20 data flows. In this case, each line is almost linearly decreasing when the “pause time” increases from 0 to 900 seconds.

The control overhead results for each protocol are similar to Scenario 1. AODV produced many more control packets than the other two protocols, and DSR produced the smallest number of control packets in most situations, except for a couple of points for 60 data flows, when it produced more control packets than STAR.

STAR had the lowest latency, with AODV and DSR having comparable latency results. For 20 data flows, at the point

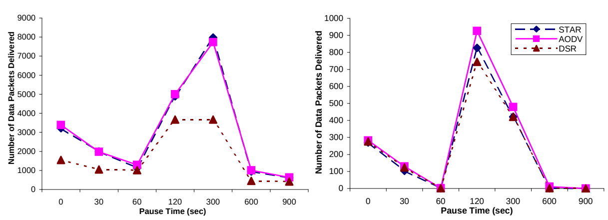

Fig. 4. Data Delivery in Scenario 2 for (a)60 (b)20 Data Flows

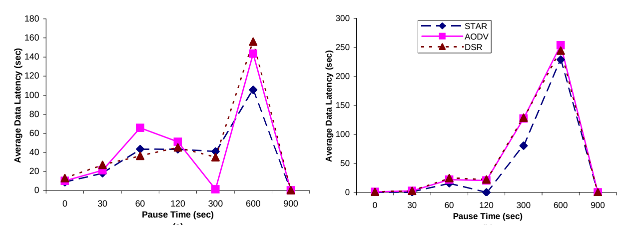

Fig. 6. Data Latency in Scenario 2 for (a)60 (b)20 Data Flows

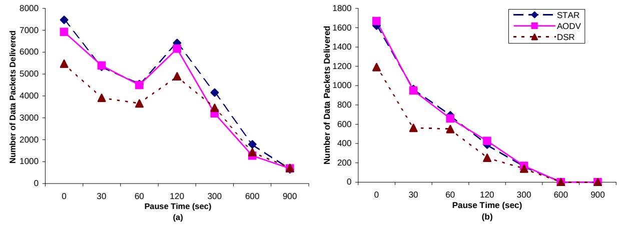

Fig. 7. Data Delivery in Scenario 3 for (a)60 (b)20 Data Flows

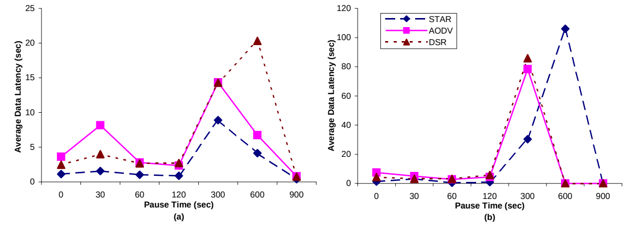

Fig. 9. Data Latency in Scenario 3 for (a)60 (b)20 Data Flows

when the “pause time” is 600 seconds, the latency values for AODV and DSR are both 0. This is the case because AODV and DSR could not deliver any data packets.

E. Scenario 4

The area used in this scenario is a 5km * 3km rectangle, and the initial node placement is random, which is the only difference of this scenario with Scenario 3.

Fig. 10, 11, and 12 show the results for this scenario.

All the three protocols delivered more data packets in this scenario than in Scenario 3, the uniform placement. STAR delivered more data packets than the other two protocols for 60 data flows, and fewer than AODV for 20 data flows. The results for STAR and AODV are comparable, and DSR delivered the fewest packets. The results in this scenario are similar with the trends of the previous three scenarios.

The results for the control overhead produced by the three protocols are quite similar to those in Scenario 1. AODV produced many more control packets than the other two protocols, and DSR produced the smallest amount. DSR produced fewer control overhead in this scenario than in Scenario 3, while the other two protocols showed comparable behavior. STAR had the lowest average latency, and AODV and DSR had similar latencies.

F. Scenario 5

Compared with the previous four scenarios, this scenario represents a more densely connected network. The area is 4km * 4km and the radio range of each node is increased to 400m, which brings denser connectivity among nodes. Fig. 13, 14, and 15 show the results for this scenario.

Concerning the data packets delivered, for 60 data flows, STAR delivered more packets than DSR, but fewer than AODV. For 20 data flows, AODV delivered the highest amount of packets, with STAR delivering the fewest. The results increase almost linearly when the “pause time” increases from 0 to 900 seconds.

STAR had the least amount of control overhead and AODV had the largest overhead. With respect to the previous scenarios, the number of control packets increased substantially for DSR, and the fluctuations for DSR are also more pronounced.

Overall, STAR had the lowest average latency in data delivery, with AODV having the highest.

G. Results Summary

According to our simulations, in terms of a combined view of the metrics: data delivery, control overhead, and average latency, STAR has the best performance among the three in sparsely connected networks. IN densely connected networks, AODV is the best performing in terms of data delivery and STAR continues to be the best in terms of routing overhead, while delivering a smaller amount of data packets than AODV.

The control overhead produced by STAR does not change as much as AODV and DSR when the number of data flows varies. This is usually difference between table-driven and on-demand protocols. Contrary to the widely held opinion that table-driven protocols have higher control overhead than on-demand protocols, STAR produced much less control packets than AODV in most of the simulated scenarios. Another no-

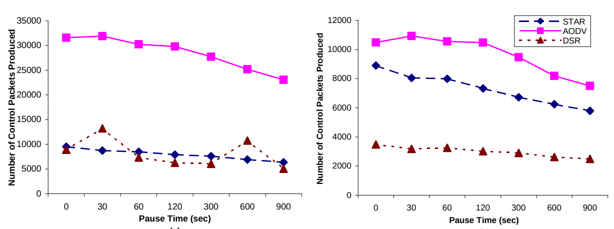

Fig. 8. Control Overhead in Scenario 3 for (a)60 (b)20 Data Flows

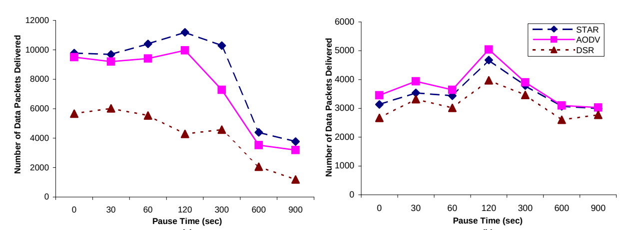

Fig. 10. Data Delivery in Scenario 4 for (a)60 (b)20 Data Flows

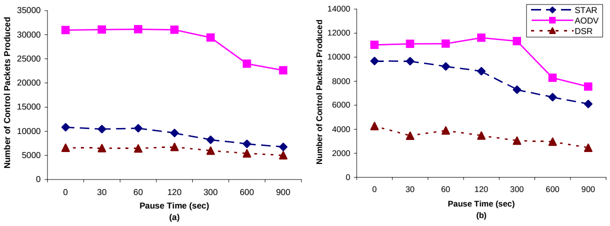

Fig. 11. Control Overhead in Scenario 4 for (a)60 (b)20 Data Flows

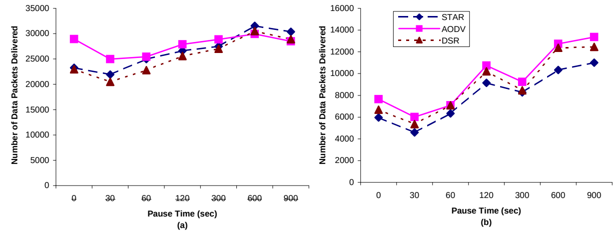

Fig. 13. Data Delivered in Scenario 5 for (a)60 (b)20 Data Flows

ticeable thing is that STAR does better in the situations with more data flows (60 flows). This result, however, should be considered in the context of the size of the ad hoc network we considered. In much larger networks, say with 200 nodes, STAR would produce much more traffic overhead. An on-demand routing protocol would be preferable in this case *if* the number of flows in the network is much smaller relative to the number of network nodes (e.g., smaller than 10% of the population). Interestingly, it appears that new solutions are needed for both types of protocols to address networks with hundreds of nodes and flows.

Under our simulation scenarios, STAR always has the lowest data delivery latency among the three protocols, this is also expected in table-driven protocols compared with on-demand protocols, because table-driven protocols discover and maintain routing information even when there is no data to be delivered.

DSR has the least amount of control overhead in sparsely connected situations; however, its data delivery rate is not satisfactory in sparsely connected situations.

H. Comparing Simulation Results under Different Simulators

These three protocols were not compared in a common simulator before, but AODV and DSR were compared using ns-2 [12], [19]. Both GloMoSim and ns-2 are discrete event simulators, with the main difference that the implementation techniques GloMoSim adopts make it more scalable and thus able to simulate larger networks [20]. For our study, which did not address very large scale networks, it was expected that the results from GloMoSim would be close to the results from ns-2 under similar scenarios. However, besides similarities, we found non-trivial differences in our results from prior studies using ns-2.

In ns-2 [19] simulations, AODV and DSR were run in densely connected scenarios. The density was higher than our scenarios.

With 10 and 20 data flows, AODV and DSR simulated in ns-2 delivered similar amounts of data, with DSR delivering slightly more than AODV. At 30 and 40 flows, AODV delivered more data than DSR, except when "pause time" is more than 600 seconds. Where the network is relatively densely connected, for our results, AODV delivered more data than DSR in most situations, but the difference is greater for higher number of data flows.

AODV produced more control overhead than DSR in ns-2 simulations, as much as five times for 40 data flows. For our simulation results, AODV also produced more control overhead than DSR, but the difference is not so big in densely connected scenarios.

AODV has lower average latency than DSR with 30 and 40 data flows in ns-2 simulations, but a bit higher latency than DSR at 10 and 20 data flows in ns-2 simulations, but higher latency than DSR at 10 and 20 data flows. This trend is not seen in our simulations. In the densely connected scenario, AODV has a higher latency than DSR in most situations.

When simulating the same protocol in different simulators, such as GloMoSim and ns-2, differences could be found in the patterns of the results. For example, in ns-2, for AODV and DSR, the control overheads increases with mobility - shorter "pause time". In GloMoSim, however, the control overhead

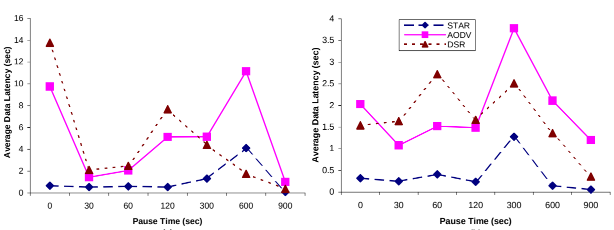

Fig. 12. Data Latency in Scenario 4 for (a)60 (b)20 Data Flows

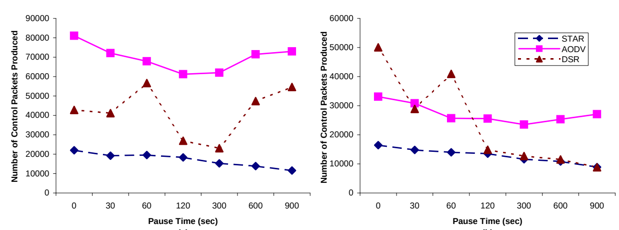

Fig. 14. Control Overhead in Scenario 5 for (a)60 (b)20 Data Flows

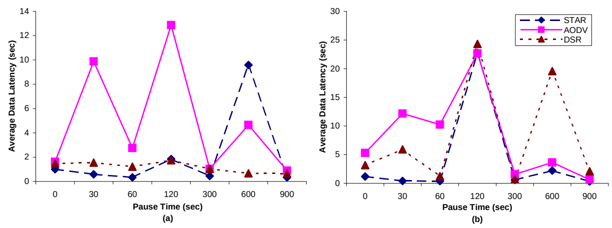

Fig. 15. Data Latency in Scenario 5 for (a)60 (b)20 Data Flows

for DSR fluctuates when the mobility decreases, while the overhead for AODV keep increasing after the point where the "pause time" is 120 seconds.

VI. Conclusions

It is not simple to determine which of the three protocols under comparison is the best for ad hoc networks. No protocol is ideal for all scenarios. A good criterion to choose a protocol might be the size and expected traffic load in the target network. Table-driven protocols like STAR face scaling problems as the number of nodes in the network grows much larger than the network size considered in this study, because the overhead traffic grows linearly with the number of destinations. On-demand routing protocols face scaling problems when the number of nodes is large and each such node has a good likelihood of contacting several other nodes in the network, because the overhead grows linearly with the number of active destinations.

According to our simulation study, in small networks (40 nodes or so) STAR performs best in sparsely connected networks, and in densely connected networks, AODV delivers more packets than the other two protocols, while STAR incurs less overhead than the other protocols. STAR always has the lowest latency, which is interesting from the fact that routes exist without the need for on-line searches of such routes. The control overhead of STAR does not change much when the number of data flows changes. The control overhead for AODV and DSR drops substantially when the number of data flows decreases.

We found similarities in the results from prior simulation studies using ns-2 as well as differences. This indicates that the simulation results serve as a good reference for studying protocol features and for comparing different protocols, but are not accurate enough for deriving conclusions about the expected performance of a given protocol in a real network.

References

[1] GEDOC Mobile Ad Hoc Networks Study Group, "What Ad Hoc Networks are," *"http://www.siam.dcc.ufmg.br/gedoc/"*.

[2] J. T. Moy, *OSPF Anatomy of an Internet Routing Protocol*, Addison-Wesley, Massachusetts, 1997.

[3] C. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Proceedings of the SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pp. 234–244, August 1994.

[4] J. J. Garcia-Luna-Aceves and M. Spohn, "Source-Tree Routing in Wireless Networks," *Proceedings of 7th International Conference on Network Protocols*, 1999.

[5] C. Perkins, "Ad Hoc On Demand Distance Vector (AODV) Routing," *Internet-Draft, draft-ietf-manet-aodv-00.txt*, November 1997.

[6] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing*, pp. 153–181, 1996.

[7] V. Park and M. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *Proceedings of IEEE INFOCOM 97*, April 1997.

[8] S. Murthy and J. J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *ACM Mobile Networks and Applications Journal*, 1996.

[9] J. J. Garcia-Luna-Aceves et al., "Wireless Internet Gateways," *Proc. IEEE MILCOM'97*, November 1997.

[10] J. Broch, D. A. Maltz, David B. Johnson, Y. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98*, pp. 25–30, October 1998.

[11] E. M. Royer and C. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Communications*, pp. 46–55, April 1999.

[12] S. R. Das, C. E. Perkins, and E. M. Royer, "Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks," *Proceedings of the IEEE Infocom*, pp. 3–12, March 2000.

[13] J. Raju and J. J. Garcia-Luna-Aceves, "A Comparison of On-Demand and Table Driven Routing for Ad-Hoc Wireless Networks," *Proceedings of ICC'00*, 2000.

[14] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90–100, February 1999.

[15] X. Zeng, R. Bagrodia, and M. Geria, "GloMoSim: A Library for Parallel Simulation of Large-scale Wireless Networks," *Proceedings of the 12th Workshop on Parallel and Distributed Simulations – PADS '98*, May 1998.

[16] D. A. Maltz, J. Broch, J. Jetcheva, and D. B. Johnson, "The Effects of On-Demand Behavior in Routing Protocols for Multi-Hop Wireless Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, pp. 1439–1453, August 1999.

[17] R. Bagrodia et. al, "Parsec: A Parallel Simulation Environment for Complex Systems," *Computer*, vol. 31, no. 10, pp. 77–85, October 1998.

[18] IEEE Standard Department, "Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) Specifications," *IEEE Standard 802.11-1997*, 1997.

[19] "The Network Simulator - ns-2," *"http://www.isi.edu/nsnam/ns/"*.

[20] L. Rajaj et. al, "GloMoSim: A Scalable Network Simulation Environment," *Proceedings of MILCOM'99*, November 1999.